



**TANGO**  
Device  
Server

---

# **KiKe Kickers User's Guide**

## **KiKe Class**

---

**Revision: - Author: vedder**  
**Implemented in C++ - CVS repository: ESRF**

### **Introduction:**

Interface class for the booster extraction kickers Ki and Ke.

### **Class Inheritance:**

- Tango::DeviceImpl
  - PowerSupply
    - KiKe

### **Properties:**

<b>Device Properties</b>		
<b>Property name</b>	<b>Property type</b>	<b>Description</b>
<b>ModbusDevice</b>	Tango::DEV_STRING	The name of the modbus device to use for communication with the PLC.
<b>RegisterBaseAddress</b>	Tango::DEV_LONG	The base address for the block of registers used for this kicker power supply.

Device Properties Default Values:

<b>Property Name</b>	<b>Default Values</b>
ModbusDevice	No default value
RegisterBaseAddress	No default value

**There is no Class properties.**

**States:**

<b>States</b>	
<b>Names</b>	<b>Descriptions</b>
<b>ON</b>	Power Supply is ON
<b>OFF</b>	Power Supply is OFF
<b>STANDBY</b>	Power Supply is STANDBY
<b>FAULT</b>	Fault detected on the Power Supply
<b>ALARM</b>	Alarm detected on the Power Supply
<b>MOVING</b>	When the device is going on, it execute a sequence. During this execution, the device state is moving.

**Attributes:**

<b>Scalar Attributes</b>			
<b>Attribute name</b>	<b>Data Type</b>	<b>R/W Type</b>	<b>Expert</b>
<b>Current:</b> The powersupply current setting in amps	DEV_DOUBLE	READ_WRITE	No
<b>Voltage:</b> The powersupply voltage in volts.	DEV_DOUBLE	READ_WRITE	No
<b>CurrentSetPoint:</b> The current set value as stored in the powersupply.	DEV_DOUBLE	READ	No
<b>PulseNumber</b>	DEV_USHORT	READ_WRITE	No

## Commands:

More Details on commands....

<b>Device Commands for Operator Level</b>		
<b>Command name</b>	<b>Argument In</b>	<b>Argument Out</b>
<b>Init</b>	DEV_VOID	DEV_VOID
<b>State</b>	DEV_VOID	DEV_STATE
<b>Status</b>	DEV_VOID	CONST_DEV_STRING
<b>On</b>	DEV_VOID	DEV_VOID
<b>Off</b>	DEV_VOID	DEV_VOID
<b>Reset</b>	DEV_VOID	DEV_VOID
<b>Standby</b>	DEV_VOID	DEV_VOID

<b>Device Commands for Expert Level Only</b>		
<b>Command name</b>	<b>Argument In</b>	<b>Argument Out</b>
<b>DBG_On</b>	DEV_VOID	DEV_VOID
<b>DBG_Off</b>	DEV_VOID	DEV_VOID
<b>DBG_Reset</b>	DEV_VOID	DEV_VOID
<b>DBG_Standby</b>	DEV_VOID	DEV_VOID

### 1 - Init

- **Description:** This commands re-initialise a device keeping the same network connection. After an Init command executed on a device, it is not necessary for client to re-connect to the device. This command first calls the device *delete\_device()* method and then execute its *init\_device()* method.

For C++ device server, all the memory allocated in the *nit\_device()* method must be freed in the *delete\_device()* method.

The language device desctructor automatically calls the *delete\_device()* method.

- **Argin:**  
**DEV\_VOID** : none.
- **Argout:**  
**DEV\_VOID** : none.
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

## 2 - State

- **Description:** This command gets the device state (stored in its *device\_state* data member) and returns it to the caller.
- **Argin:**  
**DEV\_VOID** : none.
- **Argout:**  
**DEV\_STATE** : State Code
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

## 3 - Status

- **Description:** This command gets the device status (stored in its *device\_status* data member) and returns it to the caller.
- **Argin:**

**DEV\_VOID** : none.

- **Argout:**  
**CONST\_DEV\_STRING** : Status description
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

## 4 - On

- **Description:** Switch powersupply ON.
- **Argin:**  
**DEV\_VOID** :
- **Argout:**  
**DEV\_VOID** :
- **Command allowed for:**
  - Tango::OFF
  - Tango::STANDBY
  - Tango::ALARM

## 5 - Off

- **Description:** Switch powersupply OFF.
- **Argin:**  
**DEV\_VOID** :
- **Argout:**  
**DEV\_VOID** :
- **Command allowed for:**
  - Tango::ON
  - Tango::STANDBY
  - Tango::ALARM

## 6 - Reset

- **Description:** Reset the powersupply to a well known state.
- **Argin:**  
**DEV\_VOID :**
- **Argout:**  
**DEV\_VOID :**
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM

## 7 - Standby

- **Description:** Set the kicker to standby
- **Argin:**  
**DEV\_VOID :**
- **Argout:**  
**DEV\_VOID :**
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::ALARM

## 8 - DBG\_On (for expert only)

- **Description:** Send a ON command to the PLC, without any timing or state control. This is a debug function.
- **Argin:**  
**DEV\_VOID :**
- **Argout:**

**DEV\_VOID :**

○ **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::STANDBY
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING

## **9 - DBG\_Off (for expert only)**

○ **Description:** Send a OFF command to the PLC, without any timing or state control. This is a debug function.

○ **Argin:**

**DEV\_VOID :**

○ **Argout:**

**DEV\_VOID :**

○ **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::STANDBY
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING

## **10 - DBG\_Reset (for expert only)**

○ **Description:** Send a Reset command to the PLC, without any timing or state control. This is a debug function.

○ **Argin:**

**DEV\_VOID :**

○ **Argout:**

**DEV\_VOID :**

○ **Command allowed for:**

- Tango::ON
- Tango::OFF
- Tango::STANDBY
- Tango::FAULT
- Tango::ALARM
- Tango::MOVING

## 11 - DBG\_Standby (for expert only)

- **Description:** Send a Standby command to the PLC, without any timing or state control. This is a debug function.
- **Argin:**  
**DEV\_VOID :**
- **Argout:**  
**DEV\_VOID :**
- **Command allowed for:**
  - Tango::ON
  - Tango::OFF
  - Tango::STANDBY
  - Tango::FAULT
  - Tango::ALARM
  - Tango::MOVING

TANGO is an open source project hosted by :  
**SOURCEFORGE.NET**®

Core and Tools : CVS repository on tango-cs project  
Device Servers : CVS repository on tango-ds project